VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELGAVI, KARNATAKA -590 018

A Mini-Project Report

on

"PHARMACY DATABASE"

Submitted in partial fulfillment for the *"Mobile Application Development Laboratory with* Mini-Project" (18CSMP68) course of Sixth Semester of Bachelor of Engineering in Computer Science & Engineering during the academic year 2021-22.

By

P SHASHANK RAO	**4MH19CS067**
RAMYA LAKSHMANAN M	**4MH19CS078**

Under the Guidance of

Dr. Wahida Banu
Associate Professor
Department of CS&E
MIT Mysore

MAHARAJA INSTITUTE OF TECHNOLOGY MYSORE
Belawadi, S.R. Patna Taluk, Mandya Dist-571477

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Accredited By:

2021-22

MAHARAJA INSTITUTE OF TECHNOLOGY MYSORE

Belawadi, S.R. Patna Taluk, Mandya Dist-571477

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

Certified that the mini-project work entitled "**PHARMACY DATABASE**" is a bonafide work carried out by **P SHASHANK RAO** (4MH19CS067) & **RAMYA LAKSHMANAN M** (4MH19CS078) for the **Mobile Application Development Laboratory with Mini Project** (18CSMP68) of Sixth Semester in Computer Science & Engineering under Visvesvaraya Technological University, Belagavi during the academic year 2021-22.

It is certified that all corrections/suggestions indicated for Internal Assignment have been incorporated in the report. The report has been approved as it satisfies the course requirements.

Signature of Guide	Signature of the HOD
Dr .Wahida Banu	**Dr. Shivamurthy R C**
Associate Professor	Professor & HOD
Dept. of CS&E	Dept. of CS&E
MIT Mysore	MIT Mysore

External viva

Name of the Examiners	**Signature with date**
1)………………………………………	
2)………………………………………	

ACKNOWLEDGEMENT

It is the time to acknowledge all those who have extended their guidance, inspiration, and their wholehearted co-operation all along our project work.

We are also grateful to **Dr. B G Naresh Kumar**, principal, MIT Mysore and **Dr. Shivamurthy R C**, HOD, CS&E, MIT Mysore for having provided us academic environment which nurtured our practical skills contributing to the success of our project.

We wish to place a deep sense of gratitude to all Teaching and Non-Teaching staffs of Computer Science and Engineering Department for whole-hearted guidance and constant support without which Endeavour would not have been possible.

Our gratitude will not be complete without thanking our parents and our friends, who have been a constant source of support and aspirations

P SHASHANK RAO

RAMYA LAKSHMANAN M

ABSTRACT

In this project medicines Name, Date and Time of the Day as input from the user andstore it in the SQLite database. This application focused on the people who forget to take medicines on time. It allows users to set an alarm along with the fields of date, time and Pharmacydescription which will allow them to set alarm for multiple medicines at different time intervals.

The user can activate or deactivate the notification accordingly. It will be sent as email or message as selected by the user. Medication reminders help in decreasing medication dispensing errors and wrong dosages. It is life-saving, money saving and time saving application which is easy to use and provides a good user interface.

People usually forget to take their Medicines on time and face problems tracking their intake habit. They sometimes take incorrect Medicines due to lack of information or misjudgment. Neither do they have a proper listing of the Medicines they are taking and when to consume them.

CONTENTS

Chapters **Page No.**

CHAPTER - 1

INTRODUCTION

1.1 Aim

To enter Pharmacy Name, Date and Time of the Day as input from the user and store it in the SQLite database. Input for Time of the Day should be either Morning or Afternoon or Evening or Night.

1.2 Overview

The category of patients involves all human beings-teachers, students, businessmen, housewives, children and also all of us have a busy hectic schedule. Today's life is full of responsibilities and stress. So, people are prone to diseases of different types and it is our duty to make ourselves stay fit and healthy. If the patient stays at home, then he or she might get someone to look after him/her but when one is not at home, is out of the city or state away from home then it is hard for the family members to call them and remind them their dosage timings every time.

1.3 Outcome

Many of these systems require special hardware devices to remind the patients about the Pharmacy in- take timings. Purchasing new hardware devices becomes costly and more time and money consuming. So, in the given work an attempt has been made to implement a system which is economical, easily accessible and improves medication adherence.

CHAPTER - 2

DESIGN AND IMPLEMENTATION

2.1 Algorithm

1.START

2.Click on Insert data button.

3.It shows Pharmacy name, date & time of day and fill the following

information.4.Click on Fetch data button.

5.It shows date and time of day whether time of day is morning, afternoon, evening, night.

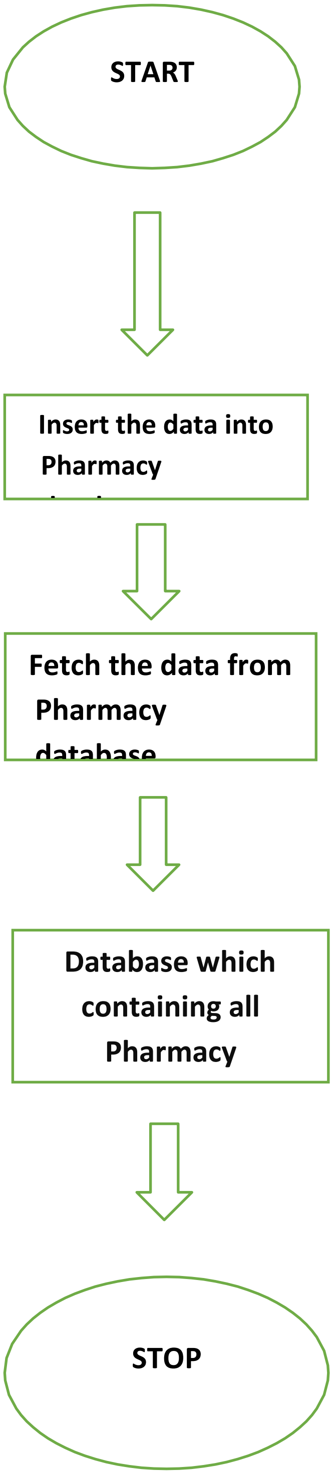
6.And retrieve the following data related to Pharmacy

database.7.Repeating the same step insert and fetch data

accordingly.

8.Quit (Stop)

2.2 Flow chart


START
Insert the data into Pharmacy
Fetch the data from Pharmacy database
Database which containing all Pharmacy
STOP

2.3 Proposed System

The proposed system is based on Android Operating system which will remind the users to take Pharmacy on time through notification. Android is a Linux-based operating system designed primarily for touch screen mobile devices such as smart phones and tablet computers, developed by Google in conjunction with the Open Handset Alliance. Android was built from the ground-up to enable developers to create compelling mobile applications that take full advantage of all a handset has to offer. The system is specified on android operating system only because the market share of Android is high. Android also comes with an application development framework (ADF), which provides an API for application development and includes services for building GUI applications, data access, and other component types. The framework is designed to simplify the reuse and integration of components.

2.4 IMPLEMENTATION

XML:

Extensible Markup Language (XML) is a markup language and file format for storing, transmitting, and reconstructing arbitrary data. The main purpose of XML is serialization, i.e., storing, transmitting, and reconstructing arbitrary data. For two disparate systems to exchange information, they need to agree upon a file format. XML standardizes this process. As a markup language, XML labels, categorizes, and structurally organizes information. XML tags represent the data structure and contain metadata. What's within the tags is data, encoded in the way the XML standard specifies. An XML document is a string of characters.

The characters making up an XML document are divided into markup and content, which may be distinguished by the application of simple syntactic rules. XML has come into common use for the interchange of data over the Internet. Hundreds of document formats using XML syntax have been developed, including RSS, Atom, and XHTML. It is the message exchange format for the Asynchronous JavaScript and XML (AJAX) programming technique.

JAVA:

Java is a high-level, class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible. It is a general-purpose programming language intended to let programmers write once, run anywhere (WORA), meaning that compiled Java code can run on all platforms that support Java without the need to recompile.

Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of the underlying computer architecture. The Java runtime provides dynamic capabilities (such as reflection and runtime code modification) that are typically not available in traditional compiled languages. Java uses an automatic garbage collector to manage memory in the object lifecycle. The programmer determines when objects are created, and the Java runtime is responsible for recovering the memory once objects are no longer in use. Once no references to an object remain, the unreachable memory becomes eligible to be freed automatically by the garbage collector.

- **Activities:** An activity represents a single screen with a user interface, in-short Activity performs actions on the screen. If an application has more than one activity, then one of them should be marked as the activity that is presented when the application is launched, which is done in the manifest file. The project has many activities, starting from Main Activity, which is the first screen visible to the users once they open the app. The clicks on the views will lead to triggers of respectiveactivities.

- **Views:** View is the basic building block of UI (User Interface) in android. View refers to the android.view.View class, which is the super class for all the GUI components. The app contains Text View, Image View, Button, Edit Text, List View, Progress Bar, Card View which helps in achieving the flow of Activities and design the layout responsively. Text Views are used to display the patientdetails at the doctor's side. Edit Texts are used to take the details during login, register and add symptoms. Image Views are used to display the icon of the app and List Views are used to display the slots for the users with details of number of available slots for appointment, timings, and id.

- **Layouts:** Android Layout is used to define the user interface that holds the UI controls or widgets that will appear on the screen of an android application or activity screen. The app is designed using Constraint Layout and Linear Layout.

- **Intents, Resources:** The app communicates or transfers the control and information from one Activity to other Activity with the help of intents. The ID ‘s, String values, Colors, Styles, XML file for every Activity, Drawable such as icons are stored in the resource folder.

SQLite:

SQLite is a software library that implements SQL database engine that is:

1.Self-contained (requires no other components)

2.Serverless (requires no server backend)

3.Zero-configuration (does not need to be configured for your application)

Transactional (changes within a single transaction in SQLite either occur completely or not at all)
SQLite is the most widely deployed database engine in the world.

The source code for SQLite is in the public domain.

SQLite Databases:

1.Stores data in tables of rows and columns.

2.The intersection of a row and column is called a field.

3.Fields contain data, references to other fields, or references to other tables.

4.Rows are identified by unique IDs.

5.Columns are identified by names that are unique per table.

It is just like a spreadsheet with rows, columns, and cells, where cells can contain data, references to other cells, and links to other sheets.

2.5 Source Code

Activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:app="http://schemas.android.com/apk/res-auto"
  xmlns:tools="http://schemas.android.com/tools"
  android:layout_width="match_parent"
  android:layout_height="wrap_content"
  android:orientation="vertical"
  tools:context=".MainActivity">


  <TextView
    android:id="@+id/textView2"
    android:layout_width="match_parent"
    android:layout_height="34dp"
    android:layout_weight="1"
    android:text="PHARMACY DATABASE"
    android:textAlignment="center"
    android:textColor="@color/black"
    android:textSize="30dp"
    android:textStyle="bold" />

  <LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="59dp"
    android:layout_marginTop="20dp"
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp">

  <TextView
    android:id="@+id/textView4"
    android:layout_width="118dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:text="Insert Data" />

  <Switch
    android:id="@+id/switcher"
    android:layout_width="96dp"
    android:layout_height="match_parent" />

  <TextView
    android:id="@+id/textView3"
    android:layout_width="122dp"
    android:layout_height="wrap_content"
    android:text="Fetch Data" />

</LinearLayout>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="59dp"
    android:layout_marginTop="20dp"
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp">


 <TextView
     android:id="@+id/txtViewMed"
     android:layout_width="122dp"
     android:layout_height="wrap_content"
     android:text="Pharmacy Name" />
 <EditText
     android:id="@+id/edtTxtmed"
     android:layout_width="wrap_content"
     android:layout_height="wrap_content"
     android:hint="Enter Pharmacy Name"

    android:layout_weight="1"/>
</LinearLayout>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="59dp"
    android:layout_marginTop="20dp"
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp">


<TextView

    android:layout_width="122dp"
    android:layout_height="wrap_content"
    android:text="Date" />
<EditText
    android:id="@+id/edtTxtDate"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:hint="Ed. DD/MM//YYYY"
    android:inputType="date"
    android:layout_weight="1"/>
</LinearLayout>
<LinearLayout
    android:layout_width="wrap_content"
```

```
        android:layout_height="59dp"
        android:layout_marginTop="20dp"
        android:layout_marginLeft="15dp"
        android:layout_marginRight="15dp">


    <TextView

        android:layout_width="122dp"
        android:layout_height="wrap_content"
        android:text="Time of Day: " />
    <Spinner
        android:id="@+id/spinner"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:ems="10"
        android:entries="@array/timeOfDay"/>

    </LinearLayout>
    <Button
       android:id="@+id/btnInsert"
       android:layout_width="match_parent"
       android:layout_height="wrap_content"
       android:text="INSERT DATA"
       android:layout_marginTop="80dp"
       android:textSize="24dp"/>

    <Button
        android:id="@+id/btnFetch"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="FETCH DATA"
        android:layout_marginTop="30dp"
        android:textSize="24dp"/>

</LinearLayout>
```

Strings.xml:

```
<resources>
   <string name="app_name">Pharmacy database lab1</string>
   <string-array name="timeOfDay">
      <item>Morning</item>
      <item>Afternoon</item>
      <item>Evening</item>
      <item>Night</item>
   </string-array>
</resources>
```

MainActivity.java:

```
package com.example.Pharmacydatabaselab1;

import androidx.appcompat.app.AppCompatActivity;

import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.CompoundButton;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.Switch;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
   EditText PharmacyName, PharmacyDate;
   TextView textViewMed;
   Spinner dayTimeSpinner;
   Switch swtch;
   Button insertButton, fetchButton;
   DBConnection dbConnection;

   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);

      dbConnection = new DBConnection(this);
```

```
    textViewMed = findViewById(R.id.txtViewMed);
    PharmacyName =
    findViewById(R.id.edtTxtmed);PharmacyDate =
    findViewById(R.id.edtTxtDate); insertButton =
    findViewById(R.id.btnInsert); fetchButton =
    findViewById(R.id.btnFetch); dayTimeSpinner =
    findViewById(R.id.spinner); swtch =
    findViewById(R.id.switcher);

    swtch.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {
      @Override
      public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {
        if (!isChecked) {
          fetchButton.setVisibility(View.INVISIBLE);
          insertButton.setVisibility(View.VISIBLE);
          PharmacyName.setVisibility(View.VISIBLE);
          textViewMed.setVisibility(View.VISIBLE);

        } else {
          PharmacyName.setVisibility(View.INVISIBLE);
          insertButton.setVisibility(View.INVISIBLE);
          textViewMed.setVisibility(View.INVISIBLE);
          fetchButton.setVisibility(View.VISIBLE);
        }

      }
    });
    insertButton.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View view) {
        String name = PharmacyName.getText().toString();
        String date = PharmacyDate.getText().toString();
        String time = dayTimeSpinner.getSelectedItem().toString();

        boolean insert = dbConnection.insertvalues(name, date, time);
        if (insert == true) {
          Toast.makeText(getApplicationContext(), "Data Inserted Sucessfully",
Toast.LENGTH_LONG).show();
          PharmacyName.setText(null);
          PharmacyDate.setText(null);


        } else
          Toast.makeText(getApplicationContext(), "Data Insertion FAILED",
Toast.LENGTH_LONG).show();

    }});
    fetchButton.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View view) {
        String date = PharmacyDate.getText().toString();
        String time = dayTimeSpinner.getSelectedItem().toString();
        String med = "";
        Cursor cursor = dbConnection.RetriveData(date, time);
        cursor.moveToFirst();

        do {
          med = med +
(String.valueOf(cursor.getString(cursor.getColumnIndex("PharmacyName"))));
          med += "\n";
        } while (cursor.moveToNext());
        Toast.makeText(getApplicationContext(), med, Toast.LENGTH_LONG).show();
      }
    });
  }
}
```

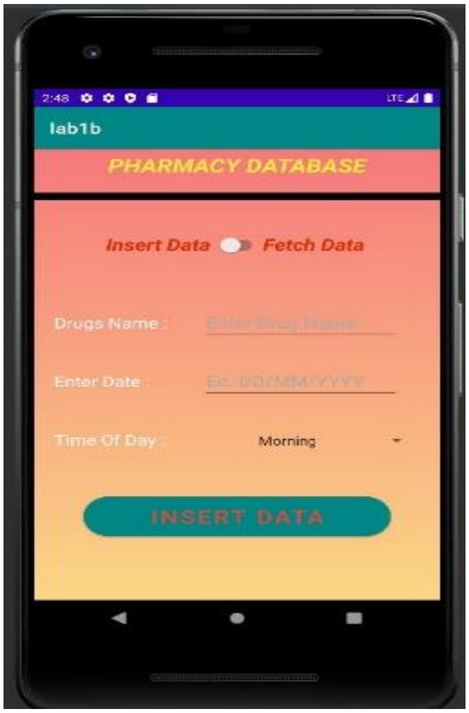
DBConnection.java:

```
package com.example.Pharmacydatabaselab1;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

import androidx.annotation.Nullable;

public class DBConnection extends SQLiteOpenHelper {
  public DBConnection(Context context) {
    super(context,"PharmacyDbase",null,1);
  }

  @Override
  public void onCreate(SQLiteDatabase dbase) {
    dbase.execSQL("create Table MedTable(PharmacyName TEXT primary key, date TEXT,time
TEXT)");

  }
  @Override
  public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {

  }
  public boolean insertvalues(String medName,String medDate,String medTime){
    SQLiteDatabase database = this.getWritableDatabase();
    ContentValues contentValues = new ContentValues();
    contentValues.put("PharmacyName", medName);
    contentValues.put("date", medDate);
    contentValues.put("time", medTime);
    long result = database.insert("MedTable",null,contentValues);

    if(result==-1)
      return false;
    else
      return true;

  }

  public Cursor RetriveData(String date, String time){
    SQLiteDatabase database = this.getReadableDatabase();
    Cursor cursor = database.rawQuery("Select * from MedTable where date ='"+date+"' AND
time='"+time+"'",null);
    return cursor;

  }

}
```

CHAPTER-3

3. RESULT ANALYSIS

3.1 Snap Shots

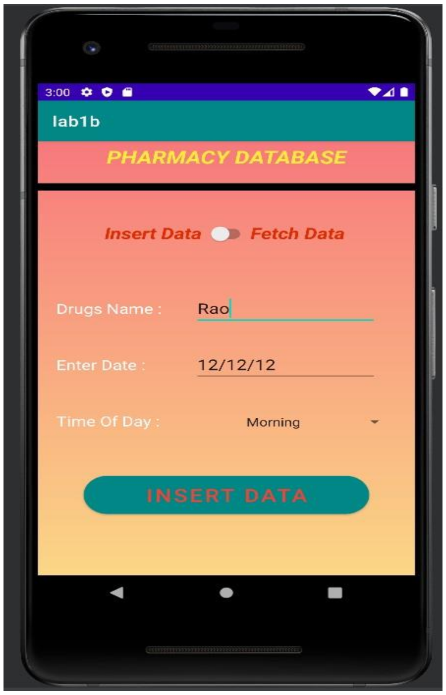


Drugs Name : Enter Drug Name

Enter Date : Ex: DD/MM/YYYY

Time Of Day : Morning

Fig3.1: Initial screen

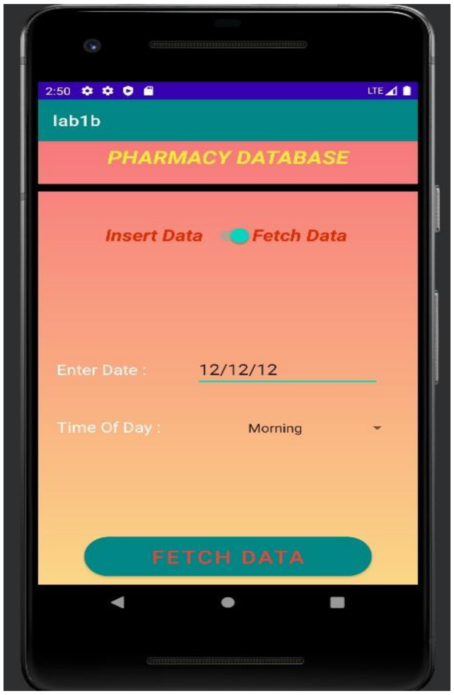


Drugs Name : Rao

Enter Date : 12/12/12

Time Of Day : Morning

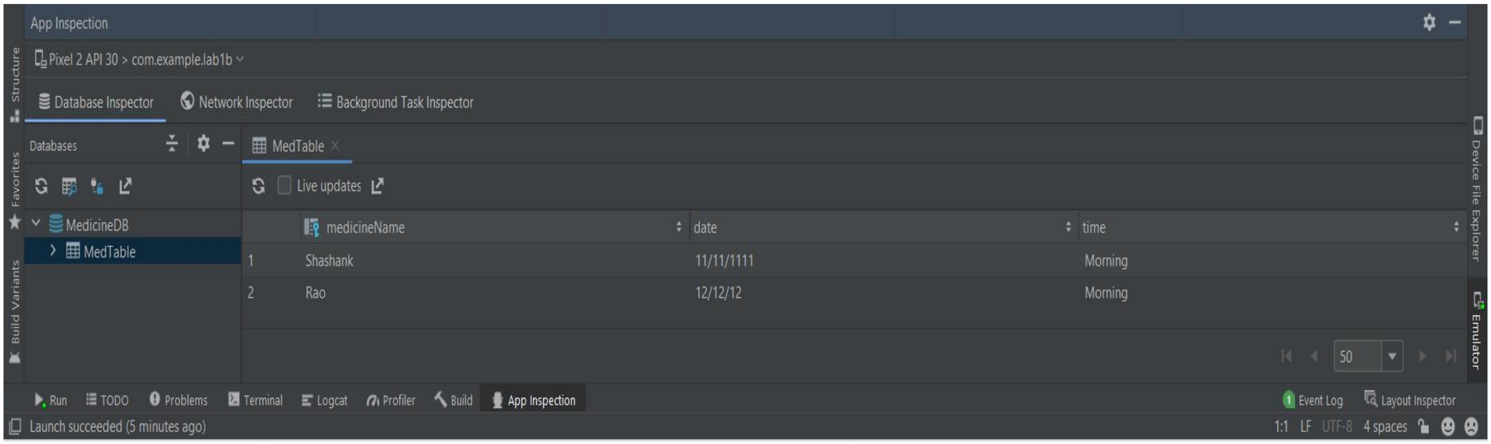
Fig3.2: Insert data into Pharmacy database



Enter Date : 12/12/12

Time Of Day : Morning

Fig3.3: Fetch data from Pharmacy database



	medicineName	date	time
1	Shashank	11/11/1111	Morning
2	Rao	12/12/12	Morning

Fig3.4: Database which containing all Pharmacy details

CHAPTER-4

CONCLUSION AND FUTURE WORK

4.1 Conclusion

Many Medication Reminder Systems have been developed on different platforms. Many of these systems require special hardware devices to remind the patients about the Pharmacy in- take timings. Purchasing new hardware devices becomes costly and more time and money consuming. So in the given work an attempt has been made to implement a system which is economical, easily accessible and improves medication adherence. Medication non-adherence reduces the effectiveness of a treatment and imposes a financial burden on health care systems. The patients will get the schedule of Pharmacy in-take time with Pharmacy description, starting and ending date of Pharmacy, notification through message or email, automatic alarm ringing system and navigation system. The scheduled reminder will not suggest any kind of Pharmacy which is not prescribed by the doctor that will assure the safety of the patient and also will avoid wrong dosages. The patientscan also search doctors' disease wise (depending upon the specialization of the doctor), which provides easy searching facility to the users and saves the time. Doctors can view all the fixed appointments along with date and time, which he fixed and through this he can make new appointment schedules. We plan to focus on improving the overall performance of the system. Also, interaction between patients and doctors through video calling and secure prescription will be focused upon. Some more ways to achieve medication adherence will be focused.

4.2 Future Enhancement

We were trying to implement one more feature which is converting back from Pharmacy database into advanced database which maintain all the Pharmacy in single stretch. But due to some technicalissues, we couldn't able to implement that feature and we will be working on it in future.

We have tried our level best to make this project very realistic so that the user does not face any trouble when switching over from any real-life android project to this highly useful one.

CHAPTER-5

REFERENCES

1. https://en.wikipedia.org/wiki/Android_Studio
2. https://en.wikipedia.org/wiki/Android_Studio#Features
3. https://developer.android.com/studio/projects
4. https://www.researchgate.net/publication/319242767_An_Android_Based_Medication_Reminder_ System_Based_on_OCR_Using_ANN
5. https://www.researchgate.net/publication/334767847_Smart_Pharmacy_Reminder_Device_For_The Elderly
6. https://www.researchgate.net/publication/321699465_Pharmacy_Reminder_for_Android_and_iOS_ users
7. https://www.sciencedirect.com/science/article/pii/S1877050916000922/pdf?md5\u003de91cf01036 8477f3d393fa07b16efb11\u0026pid\u003d1-s2.0S1877050916000922- main.pdf